3.

存储器映射 I/O（Memory-mapped I/O）是一种访问 I/O 设备的方式。在存储器映射 I/O 中，I/O 设备的控制和状态寄存器被映射到 CPU 的地址空间中的某些内存地址，程序可以直接通过访问这些内存地址来进行读写操作，而不需要通过输入输出指令来控制设备。

存储器映射 I/O 在一些情况下非常有用，特别是在需要大量数据传输的情况下。例如，图形处理器（GPU）和网络接口控制器（NIC）等设备需要大量的数据传输，因此使用存储器映射 I/O 可以更好地利用系统资源，提高数据传输效率。

4.

DMA 是“直接内存访问”（Direct Memory Access）的缩写，它是一种用于数据传输的技术。在传统的 CPU 基础上，DMA 允许外部设备通过直接和内存进行数据传输，而无需 CPU 介入。

简单来说，在使用 DMA 技术时，CPU 首先设置 DMA 控制器的寄存器，以指示需要从哪个外部设备读取数据，读取数据的起始位置，以及将数据写入到内存的起始位置等信息。然后 DMA 控制器就会自行处理数据传输的过程，包括数据的读取、写入，校验等工作，最终将数据写入到目标内存中，并通知 CPU 传输结束。

使用 DMA 技术的好处如下：

* 提高效率：由于 DMA 控制器可以直接访问内存，而不经过 CPU，因此 CPU 的负担得到减轻，系统效率得到提高。
* 减少延迟：由于 DMA 控制器可以直接访问内存和设备端口，其数据传输速度要比 CPU 快很多，因此可以减少传输的延迟。
* 解除 CPU 的限制：由于 CPU 无需直接参与数据传输，因此可以腾出更多的时间处理其他任务，提高系统的多任务处理能力。

DMA 技术在实际应用中被广泛使用，尤其是在需要大量数据传输的领域，例如音频、视频、图像处理等。DMA 可以帮助实现高速数据传输，并减轻 CPU 的负担，提高系统的性能和响应速度。

11.

（a）设备驱动程序

（b）设备无关软件

（c）设备驱动程序

（d）设备无关软件

（e）用户进程

12.

提高CPU和打印机的并行工作程序；加快进程打印输出速度，缩短进程周转时间，提高系统的吞吐。

量。

14.

如果 C 需要 S 而不是 R，这不会导致死锁。因为进程 A 已经执行完成，资源 S 可以被释放分配给 C。

如果 C 既需要 S 也需要 R 仍不会导致死锁。因为进程 A 已经执行完成，资源 S 和 R 都可以释放分配给 C，C 执行完成后，再分配给进程 B。

15.

D 再多请求一个单位，状态仍然是安全的。可用的两个单位可以先让 C 完成，然后释放

出其占用的资源，再让 D、B、A 依次完成。

C 再多请求一个单位，状态不安全。此时可用资源只有两个单位，此时 A、B、C、D 四个都不能满足要求

17.

不会导致死锁；若立即满足 A 的请求，则可用资源向量减为 A=(0020)。再将每个进程所需要的资源与可用资源向量作比较，进程 D 可以执行，执行结束后释放出它所占有的资源，进程 A、B、C、E 也可以依次执行结束。因此，所得到的系统状态仍然是安全的，不会导致死锁。

22.

多种资源的银行家算法维护了两个矩阵分别记录每个进程每种资源的已分配数量和仍需要的数量。找出可执行的进程需要将每个进程仍需要的资源数量（对应矩阵的一行）与可用资源向量进行比较，需要比较 m 次，一共 n 个进程，则需要 mn 次操作找出可以执行的进程。因为一共有 n 个进程，故需要 mn\*n= m1n2次操作。

所以 a = 1 , b = 2。

28.

(a)先到先服务。

答：选择请求的顺序为：10，22，20，2，40，6，38

磁盘臂移动的柱面数依次为：10，12，2，18，38，34，32

寻道时间：(10+12+2+18+38+34+32)×6ms=876ms

(b)下一个最邻近柱面。

答：选择请求的顺序为：20，22，10，6，2，38，40

寻道时间：(0+2+12+4+4+36+2)×6ms=360ms

(c）电梯算法（起始移动向上）。

答：选择请求的顺序为：20，22，38，40，10，6，2

寻道时间：(0+2+16+2+30+4+4)×6ms=348ms